Documentation for the swtor add-on to tails V 0.60

Authors   :   swtor00@protonmail.com
Date   :   24.10.2021
License   :   GPL 2.0

In the year 2017 I began to write the first implementation of this script for my personal usage only. This first script was one very large shell script and was running with Tails 2.2 in the background.

- This first shell-code script was growing everyday and ended in a little mess and chaos. And yes … it was ugly !
- Prior to Version 3.8 of Tails it wasn't possible to install software over the GUI. It was possible for me to implement this feature on the terminal only.
- My script was only a terminal based shell script without a fancy GUI-Menu and it was ugly to use too.
- It was not open available to the public. In the year 2018 I published the code under the GPL-licence on github.

With the current Version 0.6 of my script, the monster script that I started with Version 0.0.1 has become a bit more user friendly.

- GUI based menu-system.
- Build a local socks5 server and make ssh connection to a foreign host over the onion-network. This little cheap trick allows us to visit tor-unfriendly websites inside of Tails.
- Easy backup and restore of a complete persistent volume of a Tails system.
- Can easy be upgrade on the fly …

Have fun with my little script to enhance your user experience with Tails.

Best regards
swtor00

Table of contents

Introduction

This documentation describes how to install and use this add-on for Tails-Linux 4.23 or later. You may ask yourself, why do I need such a add-on for Tails ? The Tails Linux system already protects your privacy by using the Onion-Network on every startup in the background. It is also true that a Tails system already makes a very good job to hide your true identity and real IP-Address to the websites you are visiting. To be honest, this job is better done by the people behind Tails and their supporters that anyone could do alone by installing the TOR-Browser bundle on a regular Windows-Computer.

Although, it seems many Tor users are having difficulties surfing and navigating the regular World Wide Web (sometimes also called Clearnet Internet) , as some websites have set up discriminating rules against people who are using the Tor Browser to browse the web. At any specific moment in time when we are using Tails or the TOR-Browser in general, our personal IP packets are sent through the Internet crossing 3 different Nodes to hide the origin where the packets came from.

The so called "Exit-Nodes" of the Onion-Network can easily and instantly be detected as soon as someone queries with any remote IP Address from the list below.

https://check.torproject.org/cgi-bin/TorBulkExitList.py?ip=1.1.1.1

Therefore it is no longer a hidden secret to any visited website, that we are using the TOR-Browser or even Tails and they will often run this act of "captcha terror" against regular users of the TOR-Browser.
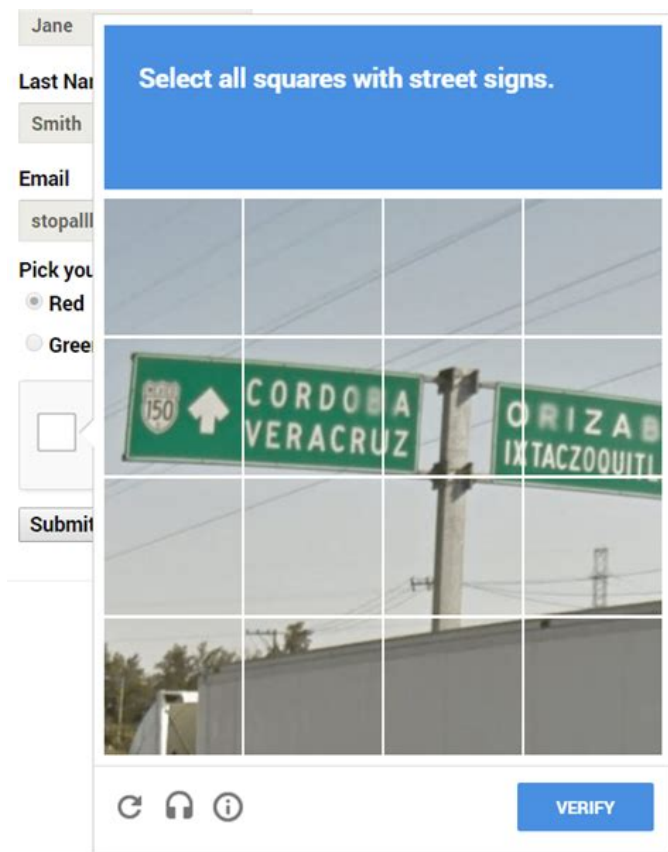
Alarmingly, the number of websites that completely block or run the "captcha-terror" against regular TOR-Users is growing in numbers day by day, to the detriment of Tor's usefulness.

Integrated inside of every Tails installation on a CD or USB medium, there is still the so called "Unsafe Browser". Because this "Unsafe Browser" doesn't make any use of the Onion-Network at all, all data communications from this browser can easily be logged by your Internet Service Provider (ISP) or your government. After Tails Version 4.8 was released,it was possible to completely prevent the starting of the unsafe Browser inside of Tails.

After Tails 4.8 was released,you see the following window on your screen by starting the "unsafe Browser".
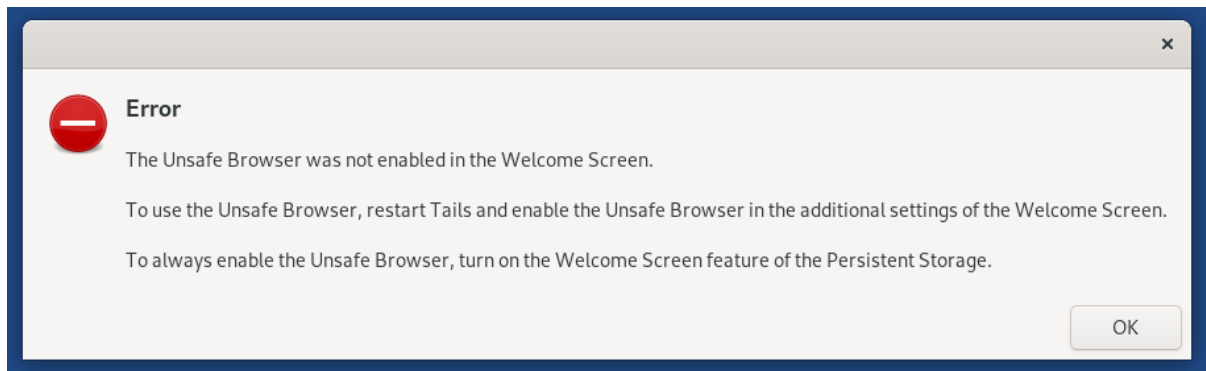
By the default settings from greeting screen of tails 4.8 or higher, the start of the "unsafe Browser" is not longer possible until a user set the default value "not starting " to "allow starting" inside of the greeting-screen of Tails.

It may make sense to use the "Unsafe Browser" inside Tails solely to browse a captive portal like it is used in many public WiFi networks for connecting, but not really for more than this simple task.

https://tails.boum.org/contribute/design/Unsafe_Browser/

Prior to activate the use of the unsafe Browser in the greeting-screen , please keep the following in mind.

IF YOU DON'T NEED THE UNSAFE BROWSER OF TAILS …
PLEASE DO NOT ACTIVATE IT IN THE GREATER-SCREEN OF TAILS
AND NEVER USE IT UNDER ALL POSSIBLE CIRCUMSTANCES !!!!

IN THE SAME MOMENT YOU ARE USING THE UNSAFE-BROWSER OF TAIL S AND VISITING A EXTERNAL WEBSITE ANYWHERE ON THE INTERNET→ YOUR REAL IP OF YOUR CURRENT USED WAN INTERFACE WILL BE LEAKED IMMEDIATELY TO THE REMOTE SITES

Some smart people even try to start a VPN connection with OpenVPN or something similar to hide the fact that they are using the Onion-Router network ,which can be blocked by any website or even an ISP.

Apart from the fact that you are only able to use a single TCP port for the very popular OpenVPN communication (all possible UDP ports inside of Tails are blocked by default), it produces many more problems than it would solve. In the endless debate about the use of a Virtual Private Network inside of Tails, I recommend to read the following URL.

https://tails.boum.org/blueprint/vpn_support/

Solution

The easiest way to solve use case 1 (which we feel is the most important one for this Tor/VPN setup) is to use a SSH connection with the `DynamicForward` option. The newly created SOCKS port can be used to have a fixed outgoing IP address. We could write on how to use that in an "unsupported, advanced users only, may kill kittens" part of the documentation.

Now, exactly right here, my special add-on for Tails enters the game called "Privacy," and provides some very useful functions for the many Tails users out there :

- Use of an encrypted SSH-connection to a Remote Host and building of a local socks5 proxy. Even the traffic that is sent over the so called "Exit-node" of our communication is still encrypted until it reaches the destination SSH-Server. When the connection packets leave the SSH-Server to any external website as a example, the packets are not longer protected by SSH and would look like a normal traffic from a standard Desktop Computer running with Linux.

- All SSH traffic is encrypted and routed over the Onion-Network, as long you are using an SSH server anywhere on the Internet. If you are using a SSH Server at your own network at home, only the connection from you tails system to the internal ssh server is encrypted. And I guess only, this is not the way you would like to go to hide the fact that you are using tails

...

- A local Browser (Chromium) with three different profiles that can be used to visit TOR unfriendly websites like Google, and many other websites that would block regular TOR users like second class Internet users. All three predefined Chromium profiles are protected against multiple actions like Web RTC and trackers in general.

- All local DNS resolution traffic on UDP Port 53 is routed over the local socks5 proxy to the remote ssh-server . This means you never contact you local DNS- Server from your current Network or ISP, like you would do it with the activated unsafe browser. And I say it again just as a reminder. Do not activate the unsafe Browser !

- For any particular website that we are visiting with Chromium anywhere on the Internet, it is no longer possible to detect that we are using the so called "Onion Router Network" to hide our personal information or even more important our current IP-Adress at all. All the traffic that the owner of the website can analyze, is coming from the regular public IP address of the remote SSH server we are currently connected to. The fact that we are using the "Evil Onion-Network" to establish our first SSH connection inside of Tails, is not visible to the websites we are visiting.

One huge problem still remains with using the Tails system to contact onion-unfriendly systems. It makes no difference what kind of bypass protocol we are using to hide the fact that we are using Tails. This so called bypass system could be done by one of the following techniques.

- A local socks5 server that connects to a remote SSH-Server

  This is the way the add-on works and is the preferred method by developers of Tails.

- Remote OpenVPN Server with a single TCP Port to connect.

  Most of all public and free OpenVPN Servers only work with a single UDP Port to connect. There are not so much Servers out there, that can be connected over TCP. Take a closer look to the following URL

  https://www.vpngate.net/en/

  This very accurate list shows multiple free OpenVPN Servers worldwide. I use this List very often to watch TV and News from a foreign Country, because every connection attempt from a foreign country is blocked inside my own country where I live(Geoblocking). I use this list with a OpenVPN Client under Linux or even Windows. But not within of Tails, for several understandable reasons. It is one things to watch only a harmless TV shows over such a VPN Server, but a completely different story to send sensitive data to a server that can be trusted for 100 %.

- Proxychains

  For my first experiments (2017) to establish a connection to a TOR unfriendly Website I was using this tool called proxychains. Sometimes it has worked for me more or less and a half hour later the program stopped working. In my humble opinion it wasn't the right tool to do the job properly. You may test it for yourself.

  https://forkdrop.io/installing-and-using-proxychains-utility-on-tails-live-boot

All the other currently wildly used VPN protocols like the following listed ones

- LL2TP
- IPSec
- WireGuard

All 3 listed protocols above are using or are depending of the UDP protocol. Inside of Tails it isn't possible to create a simple UDP connection to the direction Internet.

A big Warning about using any of the 3 presented bypass techniques including the remote ssh-servers that you are using with my own add-on here:

At this point you would like to raise awareness about the trust you are willing to give to a foreign host system and his administrators or users, who could easily read your complete communication that would be sent through to the foreign server over Tails. If you would like to use any of the 3 above bypass techniques, please don't underestimate the control they have over you in the exact moment in time that you are using their servers. Almost any VPN or SSH Service provider worldwide out there on the Internet make claims on their websites with really cheap marketing statements like the following ones:

"We don't log anything !"
"We don't spy our users !"
"We protect the privacy of our customers !"

To be honest. That's in the most cases only cheap marketing crap for foolish customers, that truly believe that marketing bullshit published by the company who offer the VPN-Service. Some every interesting articles about the so called "no-log policy" of some important worldwide VPN providers can be found right here.

http://www.theregister.co.uk/2011/09/26/hidemyass_lulzsec_controversy/

https://www.techworm.net/2018/08/reasons-why-shouldnt-use-free-vpns.html

https://www.techrepublic.com/article/16-popular-vpns-leak-your-data-heres-the-full-list/

If you read the above articles carefully, you may come to the final conclusion that you shouldn't give your personal trust out to the first person or company who offers you a VPN or SSH account for free. Even if you pay for a service with a monthly fee, there is no guarantee that you aren't being "tracked" by this VPN server or any other active users of the remote system.

If we are talking about the trust you are willing to give to a company or a remote Server, we should also talk about one of the securest email provider out there on the Internet called Protonmail that reside in Geneva / Switzerland.

The company Proton also claims that no log files are generated in general. The french customer who was arrested by the police, may see it completely different. He was using his free proton mail account over the normal url

https://account.protonmail.com/login

Proton gave the french police department the IP-Address he was using to get his personal mail. A few hours later he was arrested by the police. Be reminded again. What a company says the she would never do … They do it.

https://www.theverge.com/2021/9/6/22659861/protonmail-swiss-court-order-french-climate-activist-arrest-identification

If you are using Protonmail for Email … my personal advice is clear.

- Only use the new V3 onion address to check your Email. This means the use of Tails of course .. forget to check your Proton mail outside of Tails.

https://protonmailrmez3lotccipshtkleegetolb73fuirgj7r4o4vfu7ozyd.onion/login

To start off with, there are three simple SSH scenarios possible, in which you can use my add-on to surf the Internet without being blocked from any website you would like to visit.

Prior to show 3 possible working scenarios with my add-on, we have to talk about the dangers using it at all. From all the described SSH scenarios in the next chapter, you only should use the last presented scenario if ever possible.

- Never do a login with a username / email-address/ password that you ever used on the standard internet at anytime. If you need a valid Email-Address to register for a service or something similar, while you are using Tails, you have to create a single Email-Address that you would only use inside of Tails and nowhere else.

  NEVER TRY TO ACCESS OR PUBLISH THIS SPECIAL CREATED EMAIL-LOGIN TO ANYONE OUTSIDE OF TAILS !!!!!!!!

  YOU SEND NEVER A MAIL FROM YOU ORDINARY EMAIL TO THIS TAILS ONLY GENERATED EMAIL ADDRESS !!!!

  YOU SEND NEVER A MAIL FROM THE TAILS ONLY EMAIL TO ANYONE THAT YOU EVER CONTACTED BY YOUR ORDINARY EMAIL SYSTEM AT HOME OR IN BUSINESS !!!

  This kind of digital fingerprints (Emails / nicknames / passwords and many more little things) could be easily tracked back to you as a person with enough effort and time.

- Only use my script to solve problems with tor-unfriendly sites at all and if possible use every time the more secure builtin TOR- Browser of Tails to connecting it.

- Never using any website simultaneously inside of the TOR-Browser and the Chromium Browser used by the script.

If you really care about the above notes, you are on the save side.

1.0 Using your own SSH server inside your own Network at home

For this simple and not really recommend scenario, you need at minimum, a second computer with Linux or Unix running on your own network at home. For this SSH-daemon you could use, for example, a simple Raspberry-Pi, or of course, any other computer with an SSH daemon would work as well.
This could be implemented with a Linux System like Debian or many others without any problem. For a simple example to build a SSH-server on a Raspberry-Pi I would recommend the following URL.

https://www.raspberrypi.org/documentation/remote-access/ssh/
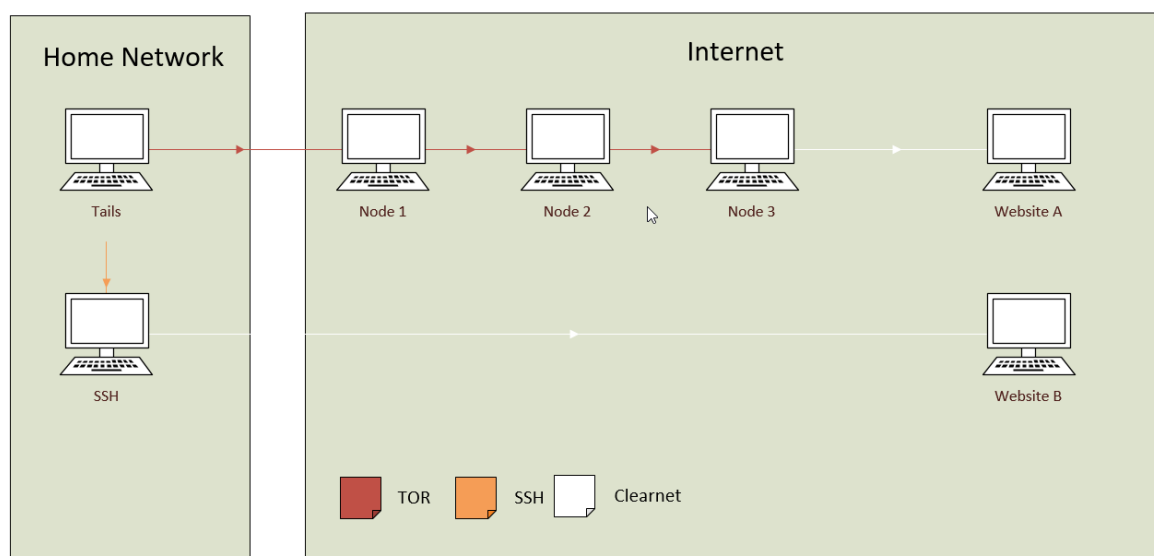
Scenario 01:

If you are using this "not so perfect scenario", you can do the following:

- Use a Website like Google or many others that normally would block your connection as soon you are trying to use it inside of Tails. To be reminded again !

  The complete Internet traffic (Data) that you send via my script through the Internet (from the SSH-server at your home to Website B for example → the white line in the graphic), can be tracked and analyzed by your current ISP, because it would be coming from a regular computer inside your own home network ! Only the websites that you are visiting with the TOR Browser over the Onion-network are secure to visit without to being tracked (Website A for example → the red line in the graphic). After the encrypted Data I leaving the exit-node of the onion-network, the network traffic is show again in the white color.

- Because the SSH-Server is inside the same physical network, the data you are sending from tails to this server aren't crossing the router into the direction Internet. These packets are not routed over the 3 nodes, because this network traffic is local.

2.0 Using your own SSH-server that can be used when you are not at home

Scenario 02
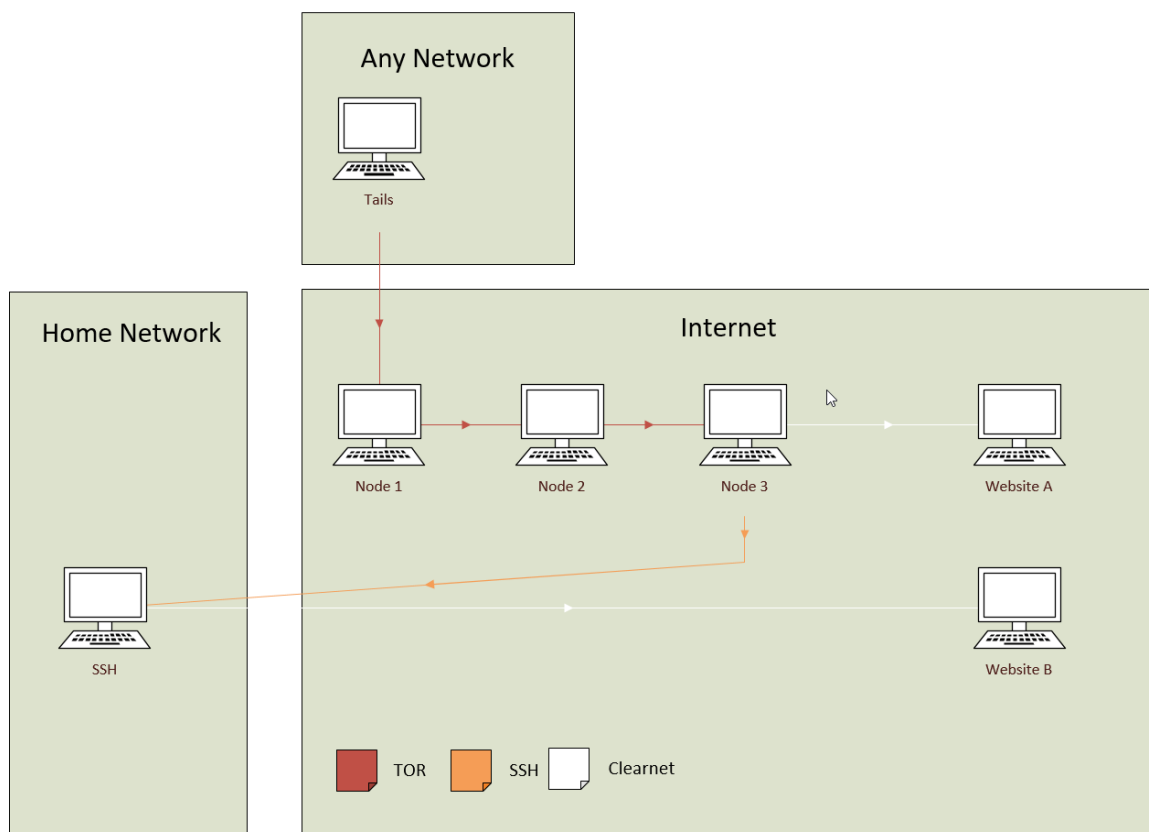
If you would like to connect to the home ssh server externally from the Internet with Tails, there is some additional work to do.


- Port Forwarding of TCP port 22 (or any other desired port you would like to use) to the destination IP inside the home network needs to be enabled. This has to be done inside your router or firewall, depending what device you are using to connect to the Internet. Most users own a router for connecting to the Internet.

- Do not allow Root Logins over SSH (PermitRootLogin no)

- Allow only a single user to login over SSH (AllowUsers username)

- Enable a DYNDNS name for your WAN IP, because most ISP's don't provide fixed IP-Addresses for the customers WAN interface. If you need a DYDNS name for your connections to the home network, I would call it a security hole that your often not aware of.

- As soon this home system is reachable from the Internet, special considerations need to be applied for the security of our SSH daemon at home.

  - You may not use the standard TCP port 22. A very good replacement port would be TCP port 443 or 53.

  - Use a key instead of a password.

  - Disable all password logins after successful logins with a key. Or would you like to have not so nice intruders from china and russia
  - that are trying to guess your root password the hole day ? This is specially true if you are using the standard port 22.

  - Put the SSH-daemon on a schedule,if you know you'll want it before hand.

  - Fake the login message to mislead the snoopers or any disliked person.

  - Only use SSH-V2, The older protocol V1 shouldn't be used anymore.

  - Disable empty passwords for all accounts.

  - Always update your system to the latest versions available because your computer could be contacted directly from the Internet.

  - Your computer at home needs to be up and running all the time if you want to contact the your home server from any location via the Internet.

I would like to emphasize the importance of the fact, like in the previous scenario, all traffic that you send out over my script can be traced by your home ISP or a local government. If you are placing a server like this in a company environment, the staff of the involved company could also tracking all your traffic that you are sending over tails with my script.

3.0 Using a remote SSH-server anywhere on the Internet (Best scenario)

Scenario 03 :
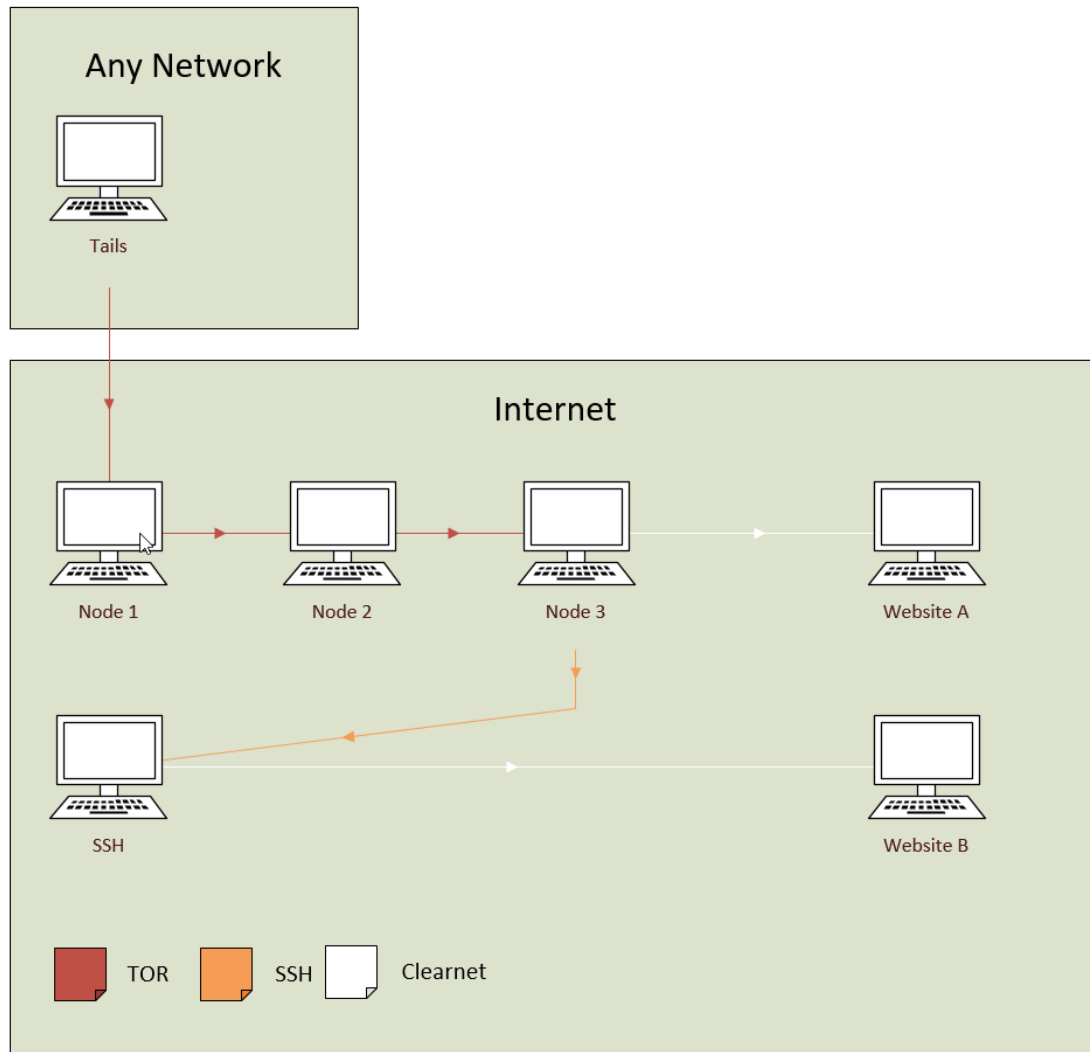
As you may now see, there are so many things that have to be configured correctly with SSH, especially if your own SSH-server can be reached from anywhere other the Internet. If you don't have a second computer inside your own network, the only suitable solution would be to find an SSH-provider anywhere on the Internet. This third option is the best option for all possible solutions to build your SSH-connection externally from Tails.

The difference from this Scenario compared to the two previous presented scenarios, is that your currently used ISP can only see the TOR traffic to the Internet. To be a bit more precise, your current used ISP can only track the connection made to the first Node (Node 1) of the chain.

Every TCP packet you send to through the Internet passes through 3 different nodes until it reaches the desired destination. During this even the complete communication between the Node 3 (Exit-Node) and the foreign SSH server is encrypted until your packets have left the SSH-server. With all the above described scenarios , one thing should highly emphasized.

You can't hide the fact, from your currently connected ISP, that you are using the TOR-Browser or even Tails itself. If you really want to hide the fact that you are using TOR or Tails, you have to do that at your own risk and against the wise advice from the creators of Tails.

There are 2 possible ways to do this.

- **Start tails in "bridge-mode"**
  https://tails.boum.org/doc/first_steps/startup_options/bridge_mode/index.en.htm

   Or

- **Use a ssh connection over TCP Port 443.This Port is used by default by websites with enccyption (https).**

So where should you start looking for a public SSH-server on the Internet ? As a good recommendation and starting point, you should have a look at the following URL :

https://shells.red-pill.eu/

Once again, we emphasize that you should only visit these multiple freeshell websites in the above link with the Tor-browser of Tails. If you are visiting them with a normal browser on Linux or Windows, you may have done a step to much to hide your personal information.

Another piece of advice from me is to use a fake-email address to register for a SSH-service of your choice. I'm sure you will find many SSH-providers on that list that meet some or all of your current requirements for a good SSH-provider. And as a third and last piece of important advise from me, never create a username for a login that could be traced back to you.

- Some of them are free of charge, others are not.

- The process to create a valid account depends from server to server.
  - The only thing needed to create a account is a e-mail account.
  - A Email and a written postcard.
  - A little riddle to prove your knowledge about Linux and Unix in general.

- Some of them don't ask for personal information about you, others would like to know almost everything about you. My simple advice use Jon Miller / New York / USA as provided contact information.

- Not so many providers from that daily growing list, give a full shell-account including a new email address or the option for X11 forwarding.

- A few Servers only allow the creation of a socks5 server. No interactive shell (like bash) of any kind.

- Many of them provide a little space to host a small website on the server .

- A few of them have databases like mariadb or mysql that you can use.

- The provided disk space to store personal files is very often limited to between 20 MB or even less. I know only one SSH-Provider of that list that has a generous size of 20GB for a single user.

- Some of these listed SSH-servers would work very well, as long you aren't trying to connect to them over the onion network. At the same moment you try to login over a public exit node from the onion network to that SSH-server, they terminate the connection immediately.
  Like the tor unfriendly websites we are already talked about, this servers are aware of the fact, that we are connecting over the ONION-network.

But wait, Yes … there is a nice and clever solution for this little handicap. First we make a SSH-connection to a server that allows us to connect over the onion-network. From our first SSH "Jumpserver" we make the desired connection to our second SSH-server. The second server can not knowing about our first ssh-connection, that we made over Orion-network . Not very easy to configure, but it will work.

Most shell-providers of the provided red-pill list do not allow the following "bad things".

- Only allows 1 active Connection with one registered login. Multiple connections with the same login would be detected and the user would be banned immediately after the second login appears.

- The installation of your own software or malware files is strictly forbidden.

- The use of port-scanners like nmap against other servers on the Internet is forbidden on most servers on the "Red-Pill" List.

- The use of software like "P2P" or "Torrent-clients" is strictly forbidden.

- On some servers is IRC allowed, on others completely forbidden.

- Some of these listed systems have hundreds of active users so be nice and keep in mind that there are other users as well and don't use up all the resources of a remote server. (Like CPU, Memory, Disk space or Bandwidth).

4.0 Preparations prior to use of this add-on

To run this script from version 0.60 or higher you need the following things.

- One USB drive with at least 8 GB capacity.
- A current Tails version Version 4.23 or higher. Tails have an excellent documentation for the installation process itself.

  https://tails.boum.org/install/index.en.html

- A persistent volume within Tails with this 3 following mandatory options activated. Without these 3 options enabled, the add-on will not work correctly as expected. These 3 options are needed.

There are also 5 Debian packages that the add-on needs to install once. Without them the script will not work correct. On every startup of Tails this 5 packages are reinstalled .

Because all our connections inside the add-on are created over SSH, we need this option to store our personal keys and the "known_hosts" files.

This one, is already active, as soon you create the persistent volume.

The next following persistent volume options are not really mandatory for the add-on for Tails, but I would recommend to use them as well, I would say they are "nice to have optional features".

 Browser Bookmarks

I also like to store my personal bookmarks on the persistent volume.

 Network Connections

If you are using multiple WiFi networks to connect, this is may the perfect option for you. Otherwise you have to type the password for every WiFi network you are using.

 Dotfiles

This feature is a must-have option for all Tails users, that would like to make a real persistent Tails. This means you only make certain settings once and you are freezing the current state.

The remaining 7 persistent volume options available are :

- Printers
- Thunderbird
- Last TOR-Enry node used
- GnuPG
- Bitcoin Client
- Pidgin
- Welcome Screen

If you use them or not, depends on your own personal choice depending on how you use Tails. My add-on can also backup all the files of a persistent volume if you would like to do so.

At this stage, from now on, please remember to do the following when you start Tails.

- You have to open the persistent volume on every start of Tails if you want to use the add-on. Of course, you can start Tails without the persistent volume activated, but the add-on itself and all the data that is stored on this persistent volume aren't usable, even the stored WiFi passwords aren't usable.

- If you are using special foreign chars for the password of the persistent volume you have to set the correct keyboard layout first, otherwise you will be typing the password with the default English Keyboard (USA) keyboard layout in the Tails greeting screen.
  Prior to version 4.12 it was not possible to store all the settings from the greeting-screen.

- The administration password of Tails also needs to be set on every start of Tails if you don't store your settings of greeting-screen . If you don't set an administrator password, the add-on won't be able to change the default local firewall. The script changes only one unique little setting inside the firewall of Tails.

  The changed setting of the firewall is very simple

 sudo -S iptables -I OUTPUT -o lo -p tcp --dport 9999 -j ACCEPT

  This little change allow us to build a local socks5 proxy over SSH. Without this modification, the predefined default rules from iptables would block any connection attempt made to port 9999 from the Loopback device 127.0.0.1

5.0 Installing the addon

After the creation of the persistent volume you have to reboot to make the persistent volume active. After the next startup you may change, the Language and the Keyboard layout:
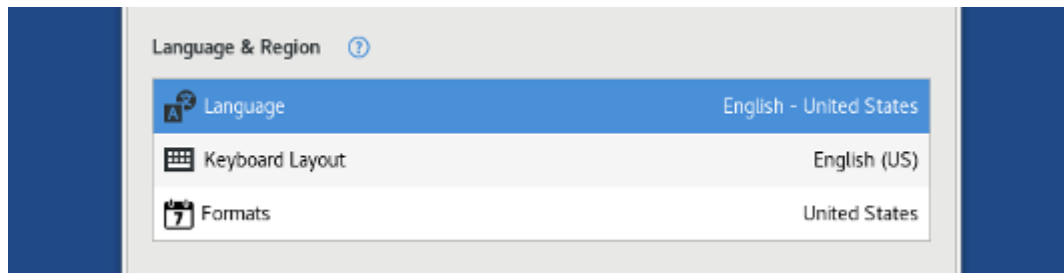
And then of course, you activate the persistent volume
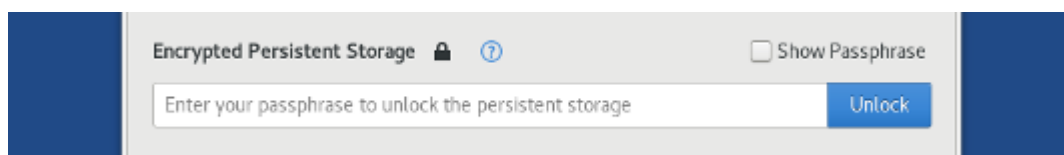
And as a last option, you set a administration password.
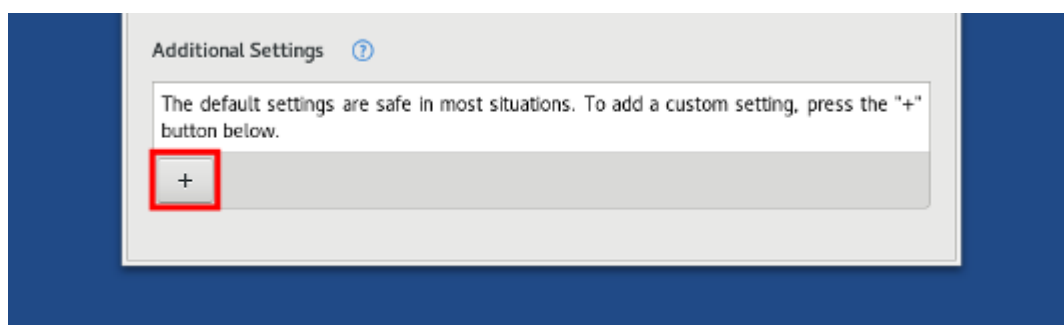
If you set the Greetings-Settings for the persistent volume active, you have to do this only once. As soon you unlock the persistent volume on the next startup, all the values you store here, are loaded again.

A little tip with the keyboard shortcuts on the startup screen of tails:

Keyboard shortcuts

Alt+L	Language
Alt+K	Keyboard Layout
Alt+F	Formats
Alt+P	Encrypted Persistent Storage
Alt+A	Additional Settings
Ctrl+Shift+A	Administration Password
Ctrl+Shift+M	MAC Address Spoofing
Ctrl+Shift+N	Network Configuration
Alt+S	Start Tails

After successfully booting Tails, please open a terminal inside the persistent folder, and then type the following command inside of the terminal.

```
amnesia@amnesia: ~/Persistent                          _  □  ✕
File   Edit   View   Search   Terminal   Help
amnesia@amnesia:~/Persistent$ █
```

git clone https://github.com/swtor00/swtor-addon-to-tails
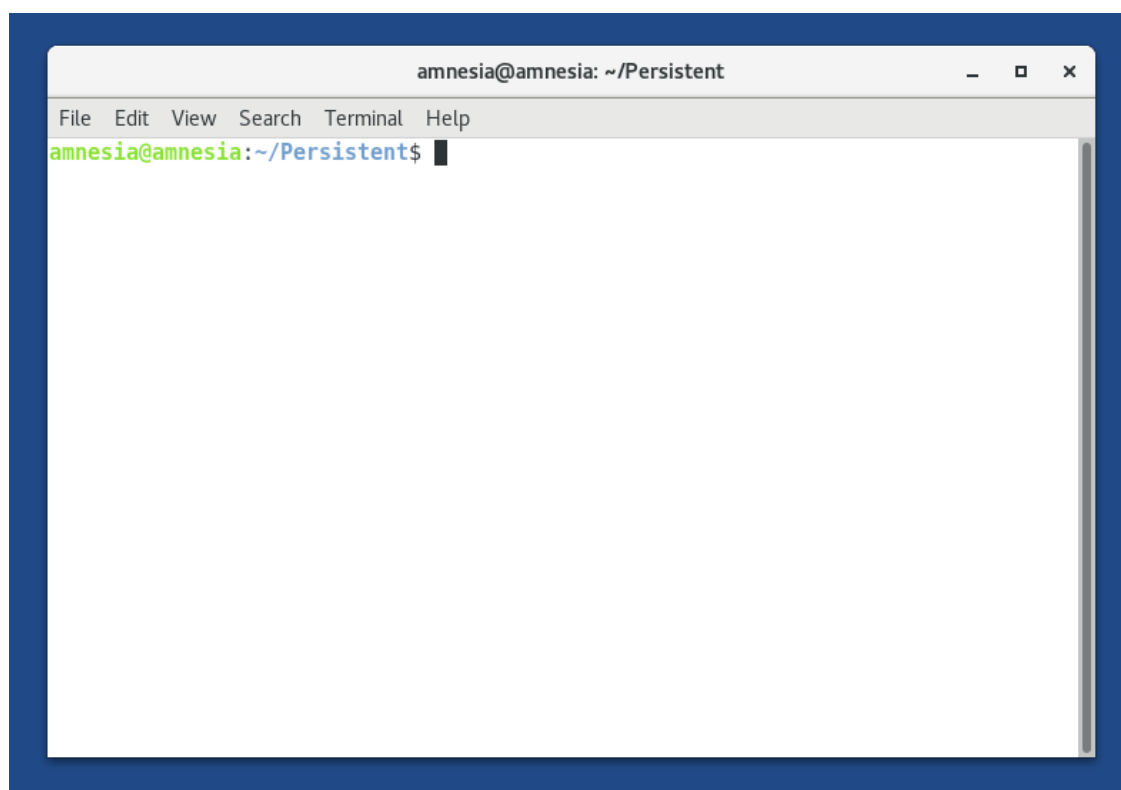
Depending on the speed of your current Internet connection, after a while you should find a folder called "swtor-addon-to-tails". Don't' worry about the current connection to github (a microsoft company since June 2018).
You are already using the onion-network to connect to the public github-server and you are not leaving any traces or evidence on the github website or that you are using this script or even Tails.

Before you do anything inside the directories of the add-on itself, you should first make a few very important decisions about the use of the add-on. The hole configuration of my add-on is written in a single file. It can be edited over the Gnome Editor (gedit) or even vim / nano if you would like to edit the file this way in the terminal. Prior to executing anything, the complete path to this configuration file is as follows.

~/Persistent/swtor-addon-to-tails/swtorcfg/swtor.cfg

The default swtor.cfg configuration file for the add-on version 0.60, looks like this.

```
SWTOR-VERSION:0.60
TAILS-VERSION:4.23
STATE: BETA
HOMEPAGE: https://github.com/swtor00/swtor-addon-to-tails
JOTV :
----------------------------------------------------------------
"Corona is a IQ test for humans ... " or even better

"Programming today is a race between software engineers striving to
build bigger and better idiot-proof programs, and the Universe trying to
produce bigger and better idiots. So far, the Universe is winning."
----------------------------------------------------------------
OPTIONS FOR THE SWTOR-ADDON

IMPORT-BOOKMARKS:NO
GUI-LINKS:YES
BROWSER-SOCKS5:YES
CHECK-UPDATE:NO
BACKUP-FIXED-PROFILE:NO
BACKUP-APT-LIST:NO
TIMEOUT-TCHECK:10
```

In the last few lines, you see all the relevant entries for my add-on. All the add-on options are written in CAPITAL letters. Let us now describe the settings possible in this file.

If you would to check for an update on every startup of the add-on, you have to replace the line containing

CHECK-UPDATE:NO with the new entry **CHECK-UPDATE:YES**

After this little change, my add-on contacts the github server that holds the master scripts on every startup to check if there is a new version to install. If you would like to manually make this update check for yourself, open a terminal and type the following commands.

---

cd ~/Persistent/swtor-addon-to-tails/scripts
./update.sh

The script will inform you if there is a new version to install. But I have to make a little warning about the use of the update-feature. This next warning applys to both of the updates possible (manual or automatic on startup).

Warning :

All local changes made to all files including the main-configuration file swtor.cfg are overwritten with the default values stored on the github server. You have to re-apply all the changes you made again ! Of course only, if you made some changes to the configuration file swtor.cfg.

If you would like to use the current used script version without the tracking and control of the git software in the background, you could execute the following terminal commands to accomplish this simple task.

```
 cd ~/Persistent/swtor-addon-to-tails
rm -rf .git
rm ~/Persistent/scripts/update.sh
```

At the same time you remove the .git directory from the add-on directory, you cannot execute ./update.sh ever again.

IMPORT-BOOKMARKS:NO

If you change the predefined value from NO to YES, the bookmarks of the TOR-Browser of my Browser are directly imported on the first startup of the add-on. The default value is NO, for one big reason. I don't like to hear, that someone's large personal bookmarks are accidental overwritten by my scripts on first startup.

GUI-LINKS:YES

If you change the value to NO, you can only start the script over a terminal. Most users should use the predefined default value YES.

BROWSER-SOCKS5:YES

Currently this entry should always be YES , In the near future it may possible with other settings like

BROWSER-PROXY / RDP-CONNECTION / VNC-CONNECTION

BACKUP-FIXED-PROFILE:NO
BACKUP-APT-LIST:NO

If you activate both options to the value YES.
What does this exactly mean to you as a user of this add-on ?

In the case you would like to backup the complete persistent volume, the size of the backup will be around 200 MB or even more. If you leave the values at the default state NO, the created backup will have the size somewhere between 3 - 5 MB.

TIMEOUT-TCHECK:10

This timeout value in seconds define, how long the scripts should wait until a error message that there is no active internet connection over the onion-network. The default value of 10 seconds should work for most users of the add-on. If your internet speed is very low, you may have to higher this value to 15 or 20 seconds.

6.0   Configuring the required SSH-connection for the add-on

For this next configuration step, we need a valid SSH account from anywhere on the Internet or our home-server. You may even try first to test this SSH connection with another operating system like Windows, Linux or even an Apple system which contains all the software needed to establish a SSH connection, generally all modern operating system … (including Windows 10 Version 1803 or higher) do have this SSH-software already included.

In my opinion, DO NOT even think about using this above discussed scenario. Under all circumstances, it's an insanely bad idea to use this SSH-connection anywhere outside of the Tails-system !! This is specially true if you are using a SSH-Server on the Internet.
If you really want to use a socks5 SSH connection to hide your browser traffic from your ISP with any other operating system than Tails, you should create a complete new login on a remote SSH-host only for that purpose !

If you would use these Tails only SSH-credentials outside of Tails, you would leak your currently used WAN IP-Address to the owner of the SSH-host immediately the moment you try to connect over SSH without the protection of the Onion-Network in the background. A simple Linux command "who | more " inside a terminal would list all the current connected users and the corresponding IP-Address from where the users are connected. An example output of the command could look this.

eao	**pts/7**	**2018-xx-23 17:27 (85.220.101.10)**
dyama	**pts/8**	**2018-xx-25 02:07 (158.3.77.185)**
tt0077	**pts/9**	**2018-xx-25 20:08 (57.41.129.24)**
jose1711	**pts/11**	**2018-xx-08 09:14 (185.177.234.117:S.0)**
piny	**pts/13**	**2018-xx-27 07:48 (237.47.33.159)**

In the perfect case scenario, that we are using the Onion-Network inside of Tails to establish the connection to our SSH-Server, the printed IP-Address for our username (column 1) in the column 4 would only be from our currently used Exit Node number 3. Otherwise it would be our real IP address from our ISP. That would be very bad. Never mix up any SSH logins credentials created only for a specific OS. Think on this twice before you do it !

An SSH account with Linux or Unix normally consists of the following information for a successful connection.

- A username like digit1 without any spaces including a valid password.
- A destination TCP port. The default port for a SSH communication is TCP port 22.
- A valid DNS-Name or a IP V4 IP-Address to connect.

All the needed configuration files for an SSH connection reside inside the directory /home/amnesia/.ssh. If this directory is empty, it means that we have never contacted any SSH-server before with our current Tails system.

To test our first SSH-Connection and see if we can successfully login over SSH, we need to open a terminal in Tails and execute the following command. You can replace the values in this example with the values provided by your own SSH-provider.

ssh -p 22 digit1@10.0.1.66

Description of the above terminal command in detail :

ssh   The local Linux command to communicate encrypted with the
                  remote SSH-Server.

-p 22   22 is the default port for a SSH connection.
                  Due to the default behavior of SSH , it isn't always necessary to add
                  -p 22 for every connection. If your SSH-providers doesn't use port
                  22, then the -p option should be used every time you invoke the
                  command.

digit1   The username for the SSH connection we are trying to establish.

10.0.1.66   The IP-Address of the remote server we would like to connect.
                  We could also provide a user friendly DNS-Name instead of a
                  IP Version 4 IP-Address.

In the case of the following scenario where we have never made an SSH active connection with our new Tails Medium including persistent volume to an SSH-server with the IP-Address 10.0.1.66, we will see a warning like following one.

The authenticity of host '10.0.1.66 (10.0.1.66)' can't be established. RSA key fingerprint is 90:8c:7d:f8:ae:1a:09:60:44:08:3b:d9:c9:f7:c4:76.

Are you sure you want to continue connecting (yes/no)?

This is the so called "public fingerprint" of the SSH-Server we are trying to connect to. The moment we type "yes" inside the terminal, the public fingerprint for this specific server 10.0.1.66 is then stored inside the file ~./ssh/known_hosts.

After storing this public key inside of Tails, on every connection we make to the SSH Server 10.0.1.66, the already stored value inside the file /ssh/known_hosts will be compared against the value that the server provides upon connecting. If there is a match of both values, we can continue to establish our secure connection.

If the two values don't match, then there is something really wrong ! If you find the following entries inside your log file or the terminal output, be very carefully with your next action. This warning sign shouldn't be ignored easily !

WARNING : REMOTE HOST IDENTIFICATION HAS CHANGED. IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY !

- It may be that someone is trying to make you think that you are connecting to the host 10.0.1.66 ,but you aren't, this is called a "man in the middle attack".

  https://en.wikipedia.org/wiki/Man-in-the-middle_attack

  If you really decide to make the connection to this host anyway, someone could now steal your current active password for that particular SSH-host or even worst, steal your current public SSH key if you ignore this very important warning and connect to this possible evil or nasty SSH-Server !!!!

- Or that the public key of the server has been replaced for some natural reason, possibly the remote server SSH-Server was replaced due to a hardware failure and the old already used SSH-Keys has never been restored.

Next, you should see the password login, for the requested user login on the remote server.You can now type the password for that account and the remote shell should appear straight afterwards.

Provided we can login to the remote server without any error, our next step would be to make this login password free in the future. We can close the remote shell on the remote server by typing "exit" or using the keyboard shortcut <ctrl><d>. Of course you could also customize a few things like changing the current password or a few other things also. If you change the current password, please write it down somewhere and store it in a safe place. If you have to start with a new empty Tails (Clean Tails Clone) , you may need to provide the password for the first SSH-Connection, to transfer the backups you made to your new installed Tails USB stick.

Our next terminal command is used to create the private / public key pair for all future SSH-communication inside of Tails. The command to accomplish this, is the following one :

```
ssh-keygen -t rsa -b 4096
```

After a short initialization time to generate these public and private keys , we have an output like the following one. This is now our own personal "holy-grail" of encrypted communication for use inside of Tails and should be saved on a regular basis.

```
                          amnesia@amnesia: ~/.ssh              _   □   ✕

 File   Edit   View   Search   Terminal   Help
drwx------ 32 amnesia amnesia  840 Oct 21 12:43 ..
-rw-r--r--  1 amnesia amnesia 1252 Jun  7 10:36 known_hosts
amnesia@amnesia:~/.ssh$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/amnesia/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/amnesia/.ssh/id_rsa.
Your public key has been saved in /home/amnesia/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:jYhowHLKvw76MjFqcia+KE3q4++PXtO0VS53+Ed+g7c amnesia@amnesia
The key's randomart image is:
+---[RSA 4096]----+
|                 |
|.                |
|o..         .    |
|o+ . . . oo .    |
|..o . ..So.+ . . |
|o.o   o o o o +  |
|.B . o o     o =.|
|@o= + .       o +|
|X%OB..         E |
+----[SHA256]-----+
amnesia@amnesia:~/.ssh$ █
```

After the creation of our personal SSH-Keys, we can copy our public key to our SSH-server. There is a special SSH command to do this. This copy command will transfer only the public Key part to the remote SSH-Server. The private key part remains in the safe hand of the persistent Volume inside Tails.

---

ssh-copy-id -i ~/.ssh/id_rsa.pub digit1@10.0.1.66

Some older Unix systems don't support the ssh-copy-id program, so with a few little bash-tricks it's possible to transfer the public key to the foreign SSH-server with the standard Unix commands every system should clearly understand.

```
cat ~/.ssh/*.pub | ssh digit1@10.0.1.66 'umask 077; \
cat >>.ssh/authorized_keys'
```

By now it should be possible, to make any desired SSH connection with Tails to the remote SSH-system 10.0.1.66 without any password or any other additional typing with the keyboard. We test the SSH connection again by typing the following command.

```
ssh -p 22 digit1@10.0.1.66
```

A full terminal from the remote host within your current terminal should appear without any password asked for the connection.

- For every single ssh-host you would like to connect to, you have to execute the command ssh-copy-id or you have to type the password again on every SSH connection you make !

- You should test every additional ssh-connection carefully so that there is no confirmation needed like adding the public key to the known_hosts file inside the directory ~/.ssh.

As soon as you have at least one SSH connection that works properly, you can create the configuration file that is needed by the add-on. Inside the "doc" directory of the add-on , you will find a small example pdf (sample-configuration.pdf) that explains exactly how to create this configuration file. The configuration of all possible SSH connections that this add-on can manage and use are defined in this single file.

~/Persistent/swtor-addon-to-tails/swtorcfg/swtorssh.cfg

SSH itself is a very complex piece of software, if you would like to have more information about SSH in general or you need some cool advanced troubleshooting tips, you should have a closer look using the TOR-Browser to navigate to the following url.

https://docstore.mik.ua/orelly/networking_2ndEd/ssh/index.htm

Some people call this reference book "The masterworks of SSH". If you have any problem related to SSH, in this book you find certain the answer.

All SSH-connections made with the add- on have a "verbose" output on all actions. You find all the verbose logs of SSH inside the following directory.

~/Persistent/swtor-addon-to-tails/swtorcfg/log